REMARKS

Claims 1-7, 9, 10-12, and 29 stand rejected under 35 U.S.C. § 101 as directed to non-statutory subject matter. Claims 1-4, 7, 10, 13, 20-22, 26, and 29-30 stand rejected under 35 U.S.C. 103(a) as unpatentable over US patent application publication 2004/0044997 by Talati (hereinafter Talati) in view of International Application WO/13003 by Rocray et al. (Rocray). Claims 5, 6, 9, 11, 12, 23-25, and 28 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Talati and Rocray in view of US patent 6,986,132 to Schwabe (hereinafter Schwabe). Claims 14-17 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Talati and Rocray in view of US Patent 6,585,659 to Hiller (Hiller). Claims 18-19 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Talati in view of Rocray, Hiller, and Schwabe.

Applicants thank the Examiner for the telephone interview of December 17, 2008. We discussed the present invention and a proposed amendment. The Examiner also suggested language for overcoming the 35 U.S.C § 101 rejection. Applicants agreed to the suggested language, and have so amended the claims along with the proposed amendment.

Response to rejections under 35 U.S.C. § 101

Claims 1-7, 9, 10-12, and 29 stand rejected under 35 U.S.C. § 101 as directed to non-statutory subject matter. Applicants have amended claims 1, 3, 10, and 29 to include the limitation of a processor executing executable code stored on a storage device as we discussed. Applicants submit that as a result of the amendments, claims 1, 2, 3, 7, 9, 10-12 and 29 are directed to statutory subject matter under 35 U.S.C. § 101. Claims 4-6 are canceled.

Amendments to the claims

Applicants have further amended claim 1 with the limitation "...a bootstrap module, within the new code image, configured to reconcile the incompatibilities by changing an initialization order and converting a format of a data structure of the old code image to a format compatible with a data structure of the new code image;" The amendment is well supported by the specification. See page 11, ¶ 41.

Claims 10, 13, 20, 29, and 30 are similarly amended. Claims 4-6 are canceled.

Claim 9 is amended with the limitation "...wherein identifying incompatibilities further~~at least one of the incompatibilities~~ comprises identifying a difference between the format of the data structure~~s~~ used by the old code image and the format compatible with the data structure~~s~~ used by the new code image." The amendment is well supported by the specification. See page 11, ¶ 40; page 19, ¶ 71. Claims 12 and 28 are similarly amended.

Applicants have added news claims 31-35. The limitations "...wherein identifying incompatibilities between the old code image and the new code image further comprises accessing capability information for the old code image and capability information for the new code image..." and "...identifying a difference between the capability information and wherein reconciling incompatibilities between the old code image and the new code image further comprises adjusting configuration settings and parameter lists..." are well supported by the specification. See page 19, ¶ 70; page 20, ¶ 74; page 22, ¶ 80. New claims 32 and 35 also include the limitations of claims 7, 19, and 26.

Response to rejections under 35 U.S.C. § 103

Claims 1-4, 7, 10, 13, 20-22, 26, and 29-30 stand rejected under 35 U.S.C. 103(a) as unpatentable over Talati in view of Rocray. Claims 5, 6, 9, 11, 12, 23-25, and 28 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Talati and Rocray in view of Schwabe. Claims 14-17 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Talati and Rocray in view of Hiller. Claims 18-19 stand rejected under 35 U.S.C. § 103(a) as unpatentable over Talati in view of Rocray, Hiller, and Schwabe. Applicants respectfully traverse these rejections.

Independent claim 1 includes the limitations:

"...a processor executing executable code stored on a storage device, the executable code comprising

a loader configured to load a new code image into a temporary memory location separate from a memory space occupied by and used by an old code image;

a logic module configured to identify incompatibilities between the old code image and the new code image from version information, **a difference in initialization requirements, and a difference in size and location between the old code image and the new code image**;

a **bootstrap module, within the new code image, configured to reconcile the incompatibilities by changing an initialization order and converting a format of a data structure of the old code image to a format compatible with a data structure of the new code image**; and

a copy module configured to copy the new code image into the memory space occupied by the old code image." Emphasis added.

Independent claims 10, 13, 20, 29, and 30 include similar limitations. Thus the present invention claims loading a new code image into a temporary memory location. The present invention then claims identifying incompatibilities between an old code image and the new code image from version information, a difference in initialization requirements, and a difference in size and location between the old code image and the new code image. The present invention further claims reconciling, using a boot module within the new code image, the incompatibilities by changing an initialization order and converting a format of a data structure of the old code image to a format compatible with a data structure of the new code image. The present invention then claims copying the new code image into a memory space occupied by the old code image.

Applicants submit that Talati and Rocray do not teach identifying incompatibilities from a difference in initialization requirements. The Examiner cites Rocray's teaching of "If such a component 42 were modified through a partial product software upgrade, the whole system would reinitialize as it would with a complete product upgrade..." as disclosing identifying incompatibilities from a difference in initialization requirements. Office Action of October 1, 2008 (OA011001), page 7, lines 1-5, citing Rocray, page 11, lines 6-8. Applicants respectfully disagree.

Rocray teaches that if a component is modified through a partial upgrade, the whole system will reinitialize just as it would for a complete product upgrade. Rocray, page 11, lines 6-8. However, there is no teaching in Rocray of identifying incompatibilities in components from a difference in initialization requirements. Rocray only uses version numbers to identify incompatibilities. Specifically, Rocray teaches that each component has a configuration data file storing a minimum configuration version. Rocray, page 8, lines 15-16. The minimum configuration version is compared with a product level version for a software release, and the minimum value must be smaller or equal to the product level version. Rocray, page 8, lines 18-20. However, Rocray, and also Talati, do not teach the element of identifying incompatibilities from a difference in initialization requirements.

Applicants further submit that Talati and Rocray do not disclose identifying incompatibilities from a difference in size and location between the old code image and the new code image. The Examiner cites Rocray's teaching of a Software Generic Control (SGC) file including a size variable as disclosing identifying incompatibilities from a difference in size and location between the old code image and the new code image. OA081001, page 7, lines 5-7, citing Rocray, page 9, Table 3. Applicants respectfully disagree.

Rocray discloses using a size and a checksum for error detection. Rocray, page 11, line 9. However, Rocray does not teach using the size to identify incompatibilities between old and new code images. As discussed above, Rocray only uses version numbers to identify incompatibilities. See for example, Rocray, page 8, lines 15-16; page 8, lines 18-20. Rocray also does not disclose using location to identify incompatibilities between the old code image and the new code image. Applicants therefore submit that Talati and Rocray do not disclose the element of identifying incompatibilities from a difference in size and location between the old code image and the new code image.

Applicants further assert that Talati and Rocray do not teach "a bootstrap module, within the new code image, configured to reconcile the incompatibilities by changing an initialization order and converting a format of a data structure of the old code image to a format compatible with a data structure of the new code image…." The Examiner cites Rocray's teaching of a boot code section and using a tag in memory to instruct a system processor to boot with alternate memory as disclosing a bootstrap module that reconciles incompatibilities by changing an initialization order as disclosing a bootstrap module reconciling incompatibilities by changing an initialization order. OA081001, page 7, lines 8-14, citing Rocray, page 15, lines 5-8; page 19, lines 6-9, 12-17. Applicants respectfully disagree.

The boot code section of Rocray does not reconcile incompatibilities between old and new code images. Rocray discloses verifying that a minimum version for a software release is smaller or equal to the product level version, and if the so, loading the product level version by copying the product level version to an alternate bank and then using the alternate bank as the active bank. Rocray, page 8, lines 18-20, page 19, lines 6-9. However, verifying that a product level version is suitable is not equivalent to reconciling incompatibilities between versions. Rocray teaches away from reconciling incompatibilities by selecting a version that is known to be compatible, such as by selecting a version with a compatible table format. Rocray, page 8, lines 13-18.

There is no teaching in Rocray of reconciling incompatibilities between old and new code images, either by changing an initialization order or by converting a format of a data structure of the old code image to a format compatible with a data structure of the new code image. Applicants therefore submit that Rocray, and also Talati, do not disclose each element of claims 1, 10, 13, 20, 29, and 30.

Because Talati and Rocray do not teach each element of independent claims 1, 10, 13, 20, 29, and 30, Applicants submit that claims 1, 10, 13, 20, 29, and 30 are allowable. Furthermore, Applicants submit that dependent claims 2, 3, 7, 9, 11, 12, 14-19, 21-26, 28, and 31-35 are allowable at least due to their dependency from independent claims 1, 10, 13, and 20. Claims 4-6 are canceled.

With regards to claims 9, 12, and 28, Applicants submit that the combination of Talati, Rocray, and Schwabe do not disclose identifying a difference between the format of the data structure used by the old code image and the format compatible with the data structure used by the new code image. Schwabe discloses discovering an incompatibility that is an attempt to store a floating point value into a field that is declared an integer type. Schwabe, col. 22, lines 30-33. Schwabe teaches verifying data structures, but does not disclose identifying incompatibilities in data structures between old and new code images. Schwabe, col. 13, lines 53-55. Therefore Schwabe, and also Talati and Rocray do not disclose the element of identifying a difference between the format of the data structure used by the old code image and the format compatible with the data structure used by the new code image, and claims 9, 12, and 28 are therefore allowable.

With regards to claims 24, 31-33, and 35, Applicants submit that the combination of Talati, Rocray and Schwabe do not teach the limitation "...identifying incompatibilities between the old code image and the new code image further comprises accessing capability information for the old code image and capability information for the new code image and identifying a difference between the capability information...." As the Examiner points out, Schwabe teaches adding functionality to a Java card after the Java card has been issued. Schwabe, col. 9, lines 6-11. However, adding functionality does not include identifying incompatibilities. Applicants therefore submit that Schwabe, and also Talati and Rocray do not teach the element of "...identifying incompatibilities between the old code image and the new code image further comprises accessing capability information for the old code image and capability information for the new code image and identifying a difference between the capability information..." and that claims 24, 31-33, and 35 are allowable.

CONCLUSION

Applicants submit that the remarks and amendments put the present application in condition for allowance. In the event the Examiner finds any remaining impediment to the prompt allowance of any of these claims, which could be clarified in a telephone conference, the Examiner is respectfully urged to initiate the same with the undersigned.

Respectfully submitted,

/Brian C. Kunzler/

Brian C. Kunzler
Reg. No. 38,527
Attorney for Applicants

Date: December 30, 2008
Kunzler & McKenzie
8 East Broadway, Suite 600
Salt Lake City, UT 84111
Telephone (801) 994-4646
Fax (801) 531-1929